

Cooperating Swarms: A Paradigm for Collective Intelligence and its Application in Finance

Sumona Mukhopadhyay
Dept of Computer Science
Army Institute of Management
Kolkata, India

Santo Banerjee
Department of Mathematics
Politecnico di Torino,
Torino, Italy

ABSTRACT

Abstract— The control of nonlinear chaotic system and the estimation of parameters is a vital issue in nonlinear science. Studies on parameter estimation for chaotic systems have been investigated recently. A variant of Particle Swarm Optimization (PSO) known as Chaotic Multi Swarm Particle Swarm Optimization (CMS-PSO) is proposed which is inspired from the metaphor of ecological co-habitation of species. The generic PSO is modified with the chaotic sequences for multi-dimension parameter estimation and optimization by forming multiple cooperating swarms. Results demonstrate the effectiveness of the scheme in successfully estimating the unknown parameters of a new hyperchaotic finance system. Numerical results and comparison demonstrate that for the given parameters of the nonlinear system, CMS-PSO can identify the optimized parameters effectively to reach the pareto optimal solution and convergence speed.

General Terms

Swarm intelligence, parameter estimation, chaos theory, Lorenz system

Keywords

Computational intelligence, particle swarm optimization, Finance system, chaos, multi-objective, global optimization.

1. INTRODUCTION

The traditional methods of optimization techniques are better suited to work on continuous and differential functions in finding the unconstrained optimal solution. But many complex real world engineering problems involve non linear, multi objective dynamical systems which are prone to get caught in areas of local optima. These have to compromise with a non optimal solution (Fig 1).

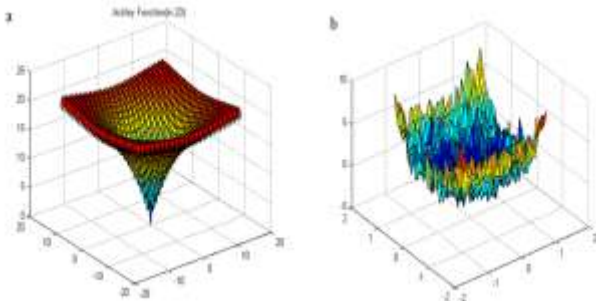


Figure 1: (a) Gradient of continuous function with single optima (b) Multi modal discontinuous functions with several optima

Meta heuristics are a class of powerful stochastic algorithms which have been well proven over the years as an efficient and fast problem solver of such magnitude. Computational Intelligence (CI) belongs to this class of Meta heuristic search technique. Meta heuristic can also be classified based on nature inspired population based search technique which include CI and a single solution based search technique like Simulated Annealing. The world of computing is facing a change in the way NP hard problems are solved. This drift in computation methodology of non linear dynamic problems in areas of engineering, bioinformatics and computing is motivated from the demand for efficient solution to problems involving high dimension in relatively faster and computational cheaper way. Researchers are gaining inspiration from the way organisms in nature solve complex problems by evolution, self learning and adaptation. These artifacts based on metaphor of biological systems have been modeled to result in systems which handle multiple variables with several candidate solutions in the presence of high degrees of non linearity.

Computational Intelligence (CI) is an emerging derivative of Artificial Intelligence which has recently gained the much deserved attention from varied fields of science, technology and management. It is a well proven meta heuristic optimization technique inspired from natural evolution, observation of behavior of biological, and neurobiological systems. Computational Intelligence spans the following listed stochastic techniques

1. Artificial Immune System [1,2]
2. Artificial Neural Network [3]
3. Fuzzy Logic [4]
4. Evolutionary Computation (EC) / Evolutionary Algorithm (EA) [5]
 - Evolutionary Strategies [6,7]
 - Evolutionary Programming [8]
 - Genetic Algorithm [9]
 - Genetic Programming [10]
5. Swarm Intelligence (SI) [11]
 - Particle Swarm Optimization(PSO)
 - Ant Colony Optimization(ACO)

The concept of SI is derived from the social interaction of organisms with their dynamic environment by a biological

process termed as stigmergy. Research is ongoing to harness the potential of bio inspired computing as computational tools in the solution of hard problems. These intelligent agents cooperate in a synchronized manner amongst each other by implicit rules of cohesion, separation and alignment focusing to solve a problem which is guided by a decision metric known as *fitness*. This is a heuristic measure which is the objective function that is to be maximized or minimized depending on the problem domain.

SI addresses the problem of global optimization(GO) by studying the adaptive systems which facilitate intelligent behavior in complex and dynamic environments where a number of intelligent organisms known as “Agents” are launched to solve the optimization problem at hand. Fig 2 shows the methodology of problem solving in the search domain of a meta heuristic population based algorithm where these intelligent agents are traversing the various sub optima in order to reach the global optima as represented in Fig 3.

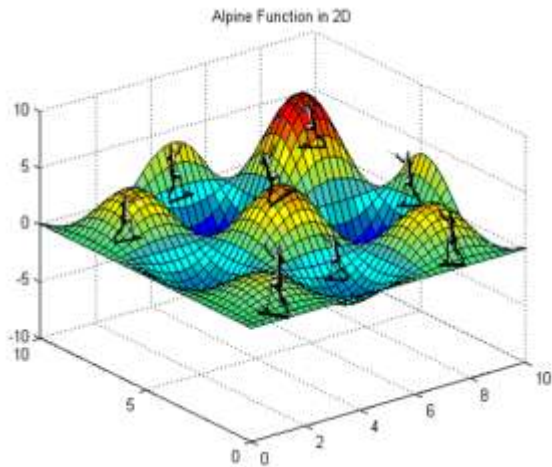


Figure 2: Multi Agent Optimization in continuous space : Randomly Initialized Agents Performing multi-agent stochastic parallel search

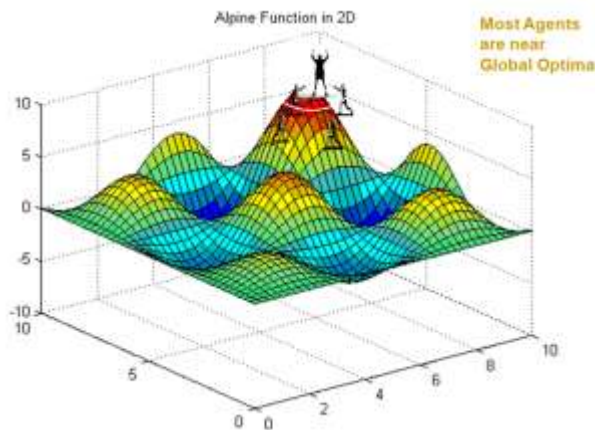


Figure 3: Multi Agents after converging at a global optima

The method was originally proposed by J. Kennedy [12] as a simulation of social behavior, and initially introduced as an optimization option in 1995. A swarm is the cluster of many

homogeneous individuals which exhibit intelligent behavior by self organizing themselves based on the exchange of information among the neighboring individuals. This branch of artificial intelligence has proved to solve many optimization problem. PSO and ACO are two popular categories belonging to the nature inspired GO techniques. Fig 4 -6 [13,14,15] captures the amazing choreography exhibited by the schools of fish, bees and flocks of birds and even micro organisms like bacteria demonstrate such an emergent behaviour. Their synchronized non collision movement is an outcome of the members of the swarm maintaining an optimum distance from their neighboring individuals. The idea behind formulation of PSO was the popular belief that social sharing of information among the individuals of a population, their flexibility and adaptability towards changes in their environment , may provide an evolutionary advantage. PSO has been proved to be an efficient method for many GO problems and in some cases it does not suffer the difficulties encountered by other EC techniques [11,16].

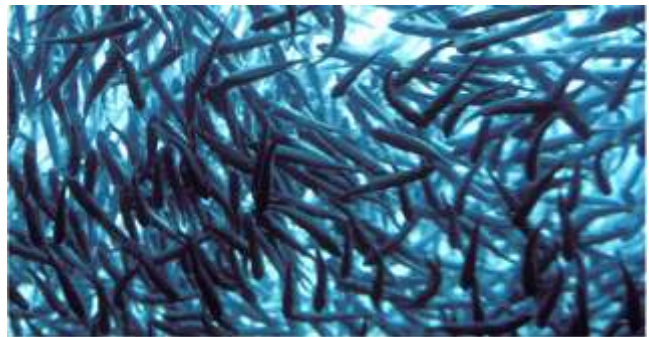


Figure 4: Swarming by Shoal of fishes – a requisite for Foraging or nesting activity



Figure 5: Swarming by Colonies of bees for bee keeping activity



Figure 6: Swarming by flock of Birds

The manuscript applies an effective cooperating multiple swarm particle swarm optimization technique injected with chaotic sequences obtained from the Lorenz system [17]. A hyperchaotic Finance model is investigated with numerical and analytical simulations. Rest of the paper is arranged as follows : Section 2 presents a background of PSO. Section 3 discusses the motivation and purpose for applying Swarm Intelligence followed by the algorithm for our methodology. In Section 4, a new autonomous hyperchaotic finance system is investigated with the experimental result. This section shows the application in global optimization and demonstrates the effectiveness of our scheme from the generic PSO followed by a conclusion.

2. PARTICLE SWARM OPTIMIZATION

The members of the swarm known as particles i fly in a d dimensional search space during their search for optimal solutions by adjusting their positions $X_i = \{x_{i1}, x_{i2}, \dots, x_{id}\}$ and velocities $V_i = \{v_{i1}, v_{i2}, \dots, v_{id}\}$. The equation of motion which governs the flight of the particles are given by

$$V_i(t+1) = \omega V_i(t) + c_1 r_1 (P_{Li}(t) - X_i(t)) + c_2 r_2 (P_G(t) - X_i(t)) \quad \text{and} \quad (1)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (2)$$

Where

$V_i(t)$ and $X_i(t)$ represent the velocity and position of particle i in the solution space at t th generation respectively. $P_{Li}(t) = \{p_{i1}, p_{i2}, \dots, p_{id}\}$ denotes the previous local position found by i within t iterations, also known as the local best position. $P_G(t) = \{p_{g1}, p_{g2}, \dots, p_{gd}\}$ denotes the global best position found among all the particles through the objective function. ω represents the inertia weight which provides a balance between global and local search. c_1 and c_2 stands for the cognitive (individual) and social (group) learning rates, respectively. r_1 and r_2 denote the uniformly distributed random numbers in the range (0,1). The parameters c_1, c_2 denote the relative importance of the memory (position) of the particle itself to the memory (position) of the swarm. Each particle flies in the search space by tuning their local position dictated by eq (1) and eq (2) to reach the goal indicated by the knowledge of the global position. A pictorial representation is shown in Fig 7.

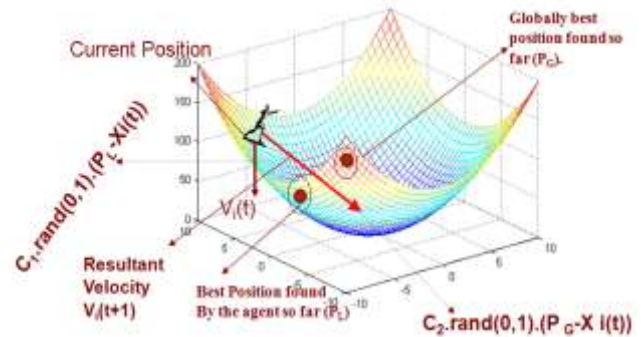


Fig 7: Particle Swarm Optimization

Their motion is decided by a heuristic metric known as fitness. The next section gives an overview of the random numbers used in general PSO scheme.

2.1 Random Number Generator System for Particle Swarm Optimization

Here we describe the system used for random number generation for the generic PSO with extremely long periods and low correlation which passed through most statistical tests.

We consider a combined multiple recursive generator [18]. The sequence can be expressed as,

$$z_n = (x_n - y_n) \bmod m_1 \quad (3)$$

where x_n and y_n are the two underlying generators with the formula,

$$x_n = (a_1 * x_{n-1} + a_2 * x_{n-2} + a_3 * x_{n-3}) \bmod m_1 \quad (4)$$

$$y_n = (b_1 * y_{n-1} + b_2 * y_{n-2} + b_3 * y_{n-3}) \bmod m_2 \quad (5)$$

with coefficients $a_1=0, a_2=63308, a_3=-183326, b_1=86098, b_2=0, b_3=-539608$, and moduli $m_1=2^{31}-1=2147483647$ and $m_2=2145483479$. In our scheme we have used the Lorenz system which serves as the random number generator $R1 = \{R1_1, R1_2, \dots, R1_i\}$ and $R2 = \{R2_1, R2_2, \dots, R2_i\}$ for every particle i in a

swarm. Thus the particles are scattered in the search space ergodically achieving enhanced population diversification but they follow an implicit trajectory which is structurally similar due to the orderly fashion of the chaotic sequences. Subsequent section elaborates on the usage of the chaotic sequences in our scheme

3. MOTIVATION AND PURPOSE

It is well established that PSO gets better results in a faster, cheaper way compared with other methods of global optimizations. Recent developments in such advanced computational methods, such as neural network and evolutionary computation have opened new doors in computational finance. In [19], the authors had effectively applied PSO to select active portfolios under a constraint on tracking error volatility. Their work considers the task of forming a portfolio of assets that outperforms a benchmark index, while imposing a constraint on the tracking error volatility. Recently Computational finance has deeply benefited from Swarm Intelligence. In 2009 the authors [20] used an interesting approach for financial classification by tapping the potential advantages of both ACO and PSO. Their mechanism assisted in the automated feature selection problem providing the decision makers with a scope to explore efficiently the solution space. The performance of their model was tested on credit risk assessment and audit qualifications financial classification problem. The proposed model was based on hybrid EA- PSO which was applied on portfolio optimization and time series forecasting. The authors also generalized the non linear fractional knapsack problem based on the observation of structural similarity present in portfolio optimization and time series forecasting problem which they implemented as the preliminary test platform for the proposed EA-PSO hybrid model. Then the authors in [21] applied PSO in the problem of single variate option pricing and compared their experimental result with standard classical Black-Scholes model for simple European options to prove the effectiveness of their scheme.

With immense development in the area of nonlinear science, the study on the complex properties of a chaotic system is becoming more intriguing and many interesting productions in nature and engineering field are yielded. In recent years, this research has been extended to complicated economy and finance systems. It is well known that economy and financial systems are very complicated nonlinear systems which are concerned with real life entities and decision making containing several complex factors. It is the developmental direction of economics to utilize the nonlinear dynamics, especially the bifurcation and chaos theory to study the internal complexity of economy and finance systems.

Multi-purpose parameter estimation methods play an increasingly important role in financial as well as insurance mathematics. Over the last few years various new derivative instruments have emerged in financial markets leading to a demand for versatile estimation methods for relevant model parameters. Typical examples include volatility, covariance and correlations. Moreover, questions of estimation concerning such extremal events have become vital in financial and insurance mathematics, for a recent detailed treatment of extreme value theory. Estimation of unknown parameters in a general is a diffusion process which is discussed in many research articles.

For continuously observed processes, a classical theory of maximum likelihood estimation has been developed including properties like consistency and asymptotic normality. In the case of discrete observations, the maximum likelihood estimator retains all the 'good' properties if the transition densities of the process are known in an explicit form. However, in most cases relevant for finance, we do not have explicit expressions for the underlying transition densities apriori, and the use of approximate likelihood functions leads to inconsistent estimators when the time between observations is bounded away from zero. There are some alternative estimation methods, especially martingale estimating functions proposed by Bibby–Sorensen [22], leading to consistent and asymptotically normal estimators.

3.1 Problem Statement:- Global Optimization and Parameter Estimation of a Hyperchaotic Finance System by Chaotic Multi-Swarm Particle Swarm Optimization

The control of nonlinear chaotic system and the estimation of parameters is a daunting task till date. Studies on parameter estimation for chaotic systems have been investigated recently [23,24]. However, a major pitfall of most global optimization problem is its tendency to get stuck in a non optimal solution. This motivated to develop efficient strategies to overcome this problem. In 2005, the authors in [25] developed a dynamic multi-swarm particle swarm optimizer (PSO) having swarms of small size and proved the effectiveness by applying it on a set of shifted rotated benchmark function. Recently, in 2010, the authors J. Sun et al [26], presented a novel Drift Particle Swarm Optimization (DPSO) algorithm, and applied it in estimating the unknown parameters of Lorenz and Chen system. Then another modified version of PSO was demonstrated by the authors [27], in the form of a parallel multi-swarm optimization (PMSO) algorithm with the aim of enhancing the search ability of the generic single-swarm PSO for global optimization of very large-scale multimodal functions.

In this manuscript, we adopt the chaotic sequences obtained from Lorenz system instead of random sequences, described in eq (3)-eq (5), as a powerful mechanism to diversify the population and improve the performance of generic PSO in preventing premature convergence to local minima. Considering a chaotic n-dimensional system given by

$$\dot{x} = f(x_1, x_2, \dots, x_n, \theta)$$

Where $\theta = \{ \theta_1, \theta_2, \dots, \theta_d \}$ are the system parameters.

To estimate the parameters we construct an estimated system described as

$$\dot{y} = F(y_1, y_2, \dots, y_n, \bar{\theta})$$

where y is the n dimensional state vector and $\bar{\theta} = \{ \bar{\theta}_1, \bar{\theta}_2, \dots, \bar{\theta}_d \}$ is a set of estimated parameters.

The multi-objective formula J for the chaotic system is given by

$$J = \frac{1}{n} \sum_{k=1}^n \|x_k - y_k\|^2 \quad (6)$$

Our task is to minimize J by searching suitable parameters which estimates θ . The modified eq (7)-(8) represents the governing equations for coordinates of i th particle in the current iteration t

$$V_i(t+1) = \omega V_i(t) + c_1 R1_i(P_{Li}(t) - \theta_i(t)) + c_2 R2_i(P_{Gi}(t) - \theta_i(t)) \quad (7)$$

$$\theta_i(t+1) = \theta_i(t) + V_i(t+1) \quad (8)$$

(a). Multi Swarm PSO (MS-PSO):-

The task of parameter estimation is converted into a global multi optimization problem. In nature, organisms split themselves into sub groups and share information about the food location or about the predator. But the sub groups have non conflicting interest. This is especially observed in the foraging behavior of bee keeping (Fig 8) with the collective goal of all swarms being to accumulate good quality honey.

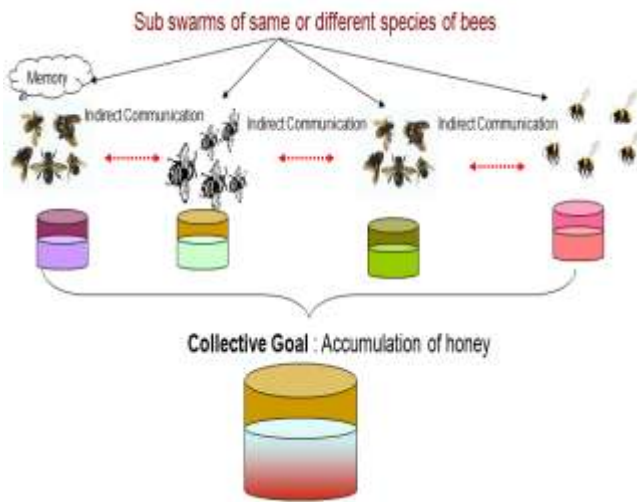


Figure 8: From Multi Swarm Foraging Technique to Multi Swarm PSO (MS-PSO)

CMS-PSO estimates more than a parameter simultaneously inspired from ecological behavior. In nature, various individuals of species co-exist and social sharing of information among the individuals of a population, may provide an evolutionary advantage. The task of global optimization of a d multi dimensional system is subdivided by d number of swarms for every parameter appearing in an n dimension state vector. Thus we have at our disposal a novel Chaotic Multi Swarm Particle Swarm Optimization (CMS-PSO) which spawns d multiple clusters of swarm S. Each swarm S_p , present in a fitness function J_p , handles p ($p \in d$), parameter and communicates values of

every member in its swarm to the other swarm which represents another unknown parameter also present in the same state vector. Thus there is a load balancing effect for evaluating the MOO. A parameter p is said to be optimized if every particle in S_p converges to the optimal value by tuning its local best position towards the global best value and the particles of the remaining ($S_d - S_p$) present in J_p would also stop moving in the solution space. So, the behavior of a swarm is governed by the estimated result of the neighboring swarm appearing in the same objective function as observed in natural case of co habitation within the same environment. Each d number of swarm is delegated with the task of optimizing the objective function. Thus a commensalism relationship is formed in which two organisms of different species biologically interact where one individual derives a fitness benefit and the other remains unaffected. This ecological facilitation achieves load balancing for the task of global optimization is delegated to swarms which are split to handle a parameter by exchanging literally information about the estimated values at every iteration. This further leads to faster computation with a scalability factor.

The multi-objective formula J for all parameters p, is said to be optimized when

$$\text{Minimize } J_p \leq \epsilon \quad (9)$$

where ϵ is an infinitesimal small positive number.

(b). Chaos and MS- PSO = CMS-PSO :-

The area for improvement has been identified by replacing the random number generator for every particle in a swarm with a chaotic sequence obtained from the Lorenz system. This has resulted in faster convergence and subtle escape from local optima. The ergodicity and mixing plays a vital role in communication as pointed out by Shannon [28]. Moreover, the dynamics of chaotic systems have been applied in various engineering applications [29,30], have demonstrated significant improvement in efficiency. The stochastic properties of chaotic systems have been effectively employed in this work also. The modified eq (7) shows that each particle in a swarm has a different random number with it obtained from the non linear Lorenz system. The individuals in a swarm have different learning schemes influenced by their different cognitive and social learning behavior. We adhere to the linearly decreasing inertia weight suggested by Shi [31] as

$$\omega = \omega_{\max} - (\omega_{\max} - \omega_{\min}) / t_{\max} * t \quad (10)$$

So, the ergodicity and randomness of the chaotic sequences transforms the search mechanism of yet another dynamic system into a stochastic process achieving exploitation and exploration of the search space.

We present the algorithm for the described process. Each particle in a swarm is heuristically initialized at first with similar velocity and system parameters within a range according to the problem, such that all the particles initially "fly" in the same trajectory. The random chaotic sequences are assigned to each particle in all the swarm. As a result, the particles are diversified in the search space preventing premature convergence to a local optima. Every

swarm estimates one kind of parameter and the knowledge or output of this swarm is learnt by another swarm of particles which have the task of estimating another parameter. Each particle can keep track of its own solution, which resulted in the best fitness, as well as keep track of the best performing candidate solution in the neighborhood of its own swarm. With this mechanism, the cluster of swarm mutually co exist in a dynamical environment with each other and share information. A schematic representation of our scheme is illustrated in Fig 9.

Schematic Diagram of Global optimization by cooperating CMS-PSO

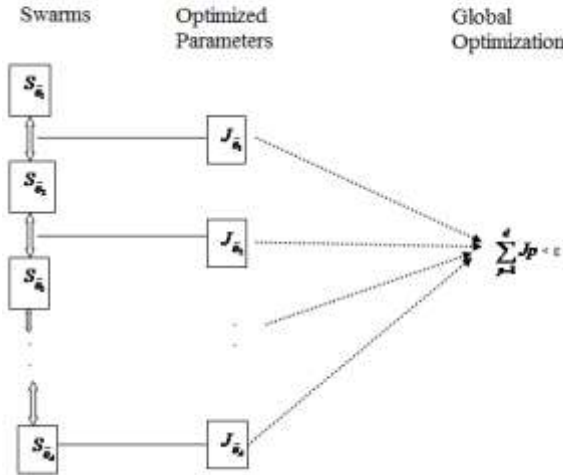


Figure 9 : Schematic Diagram of the Principle of CMS-PSO

3.2 Algorithm

1. **Initialization of vectors**:- The d swarms with fixed number of predefined particles i evolving for t number of generations are initialized randomly with velocity in the range $[V_{min}, V_{max}]$ and random numbers from Lorenz system. Generate the initial population of d number of swarms $S_{\bar{\theta}_d}$ for every $\bar{\theta}$ randomly. Predetermine which cluster of swarms will co-exist depending upon the system to be optimized.
2. **Fitness evaluation** :- Evaluate the objective function values corresponding to the particles from eq (6).
3. Find the velocities and position of particles from eq (7) and (8) respectively. All particles will be moving to the optimal point with a velocity. Set the iteration number as $t = 1$.
4. **Update CMS-PSO** :- In the t th iteration, find the following two important parameters used by a typical particle i :
 - (a) The historical best value of $\bar{\theta}_i(t)$ is $P_{Li}(t)$, for Swarm S with the optimal value of the objective function, $J[\bar{\theta}_i(t)]$, encountered by particle i in all the previous iterations. The historical best

- value of $\bar{\theta}_i(t)$ (coordinates of all particles up to that iteration t) is P_G , with the minimal value of the objective function $J[\bar{\theta}_i(t)]$, encountered in all the previous iterations by any of the i particles for the swarm .
- (b) Find the velocity of particle i in the t th iteration from eq (7). The values of c_1 and c_2 are usually assumed to be 2 so that $c_1 R_{1i}$ and $c_1 R_{2i}$ ensure that the particles would overfly the target about half the time.
- (c) Find the position or coordinate of the i th particle in t th iteration from eq (8). Evaluate the objective function values corresponding to the particles.

5. **Termination** :- The process is continued until all particles converge to the optimum solution for all swarms i.e $J \leq \mathcal{E}$.If the convergence criterion is not satisfied, step 4 is repeated by incrementing the iteration, and by computing the new values of P_{Li} and P_G .

4. THE SYSTEM AND ITS CHAOTIC PROPERTIES

In apropos to the above discussions, in the analysis presented, we study a mathematical model of a Finance system where the variables are the interest rate x_1 , the investment demand x_2 , the price index x_3 and a state feedback controller x_4 . This is a hyperchaotic finance system consisting of three differential equations with a feedback control developed in [32,33,34,35]. The task is to estimate and optimize its unknown parameters by CMS-PSO.

The system goes period doubling route towards the hyperchaotic and chaotic state for different combinations of parameter values. The nonlinear properties are studied with the help of time series of the chaotic data sets. The hyperchaotic nature is examined by the variation of the Lyapunov exponents of the system. It is observed that the system has more than one positive Lyapunov exponents for different sets of parameters. Some statistical investigations are also observed with the data output of the system.

The system (eq 9-12) is a set of four ordinary differential equations that produces hyperchaos (a chaotic attractor with more than one positive Lyapunov exponents) for a particular set of parameter values expressed as

$$\dot{x}_1 = f(x_1, x_2, x_3, x_4) = -a(x_1 + x_2) + x_4 \quad (9)$$

$$\dot{x}_2 = f(x_1, x_2, x_3, x_4) = -x_2 - a x_1 x_3 \quad (10)$$

$$\dot{x}_3 = f(x_1, x_2, x_3, x_4) = b + a x_1 x_2 \quad (11)$$

$$\dot{x}_4 = f(x_1, x_2, x_3, x_4) = -c x_1 x_3 - d x_4 \quad (12)$$

In the above equations, x_1, x_2, x_3, x_4 are the state variables and a, b, c, d are the system parameters. Now to estimate the unknown

parameters we choose $a = 3$, $b = 15$, $c = 0.2$ and $d = 0.12$ which represents the hyperchaotic mode of the system. Fig 10 represents the phase portraits of the system.

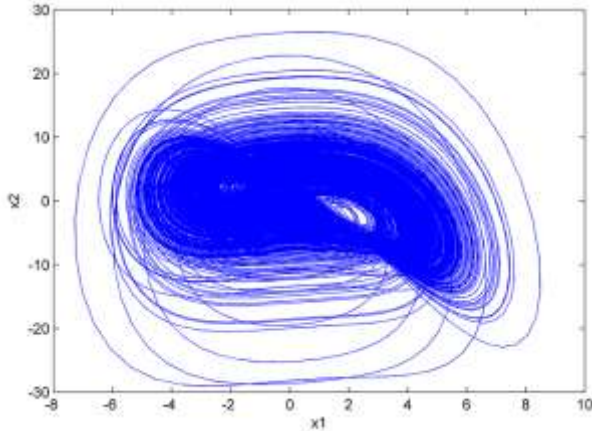


Figure 10 : Phase Space Diagram of the chaotic attractor for $a=3$, $b=15$, $c=0.2$ and $d=0.12$ parameters

In the context of multivariable optimization, the swarm is assumed to be of specified or fixed size with each particle located initially at random locations in the multidimensional design space. Each particle is assumed to have two characteristics: a *position* and a *velocity*. Each particle wanders around in the design space and remembers the best position (in terms of the food source or objective function value) it has discovered.

4.1 Result

For the sake of simplicity, parameter estimation of a two-dimensional system has been attempted. Suppose we estimate and optimize the parameters c and d .

The searching range of parameters is set as $0.01 \leq c \leq 0.5$ and $0.01 \leq d \leq 0.15$ with a population of 10 particles evolving in maximum iteration $t_{\max} = 200$ generations. The original system firstly evolves freely from a random initial state. After a period of transient process, a state is selected as the initial state y_0 for parameter estimation. The parameters of CMS-PSO like velocity are set the same randomly in the range $[V_{\min}, V_{\max}] = [0, 10]$ and $[\omega_{\min}, \omega_{\max}] = [0.4, 0.9]$ for both the swarms for the estimated system.

If we estimate parameter c and d from eq 12 of the original system then the estimated system is expressed by eq 12.1

$$\dot{y}_4 = F(y_1, y_2, y_3, y_4) = -\bar{c} y_1 y_3 - \bar{d} y_4 \quad (12.1)$$

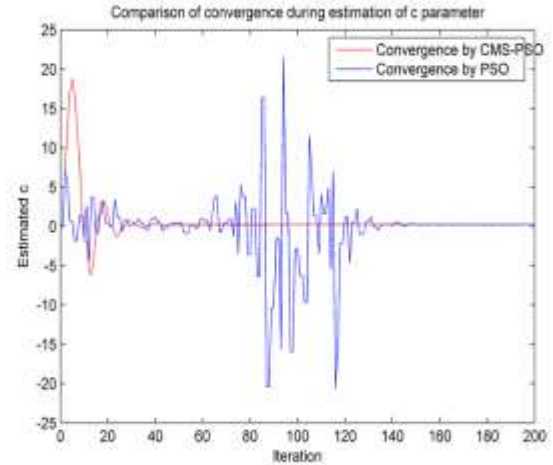


Figure 11: Comparison of convergence of CMS-PSO with PSO during estimation of parameter c

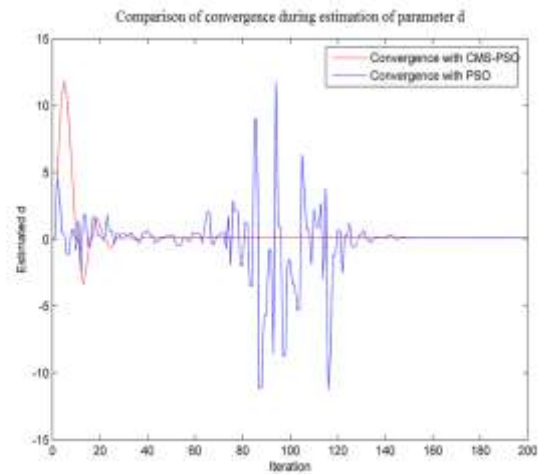


Figure 12: Comparison of convergence of CMS-PSO with PSO during estimation of parameter d

Fig. 11 and Fig 12 shows the search process during the estimation of parameter c and d respectively. Both figures highlight that the particles in PSO have a rugged gradient as compared to the proposed CMS-PSO. Also CMS-PSO converges faster and smoothly at 94 and 80 iterations for parameter \bar{c} and \bar{d} respectively. In contrast, the particle in PSO which converges first is only at iteration 190 and 195 for parameter \bar{c} (Fig 11) and \bar{d} (Fig 12) respectively with the same system setting.

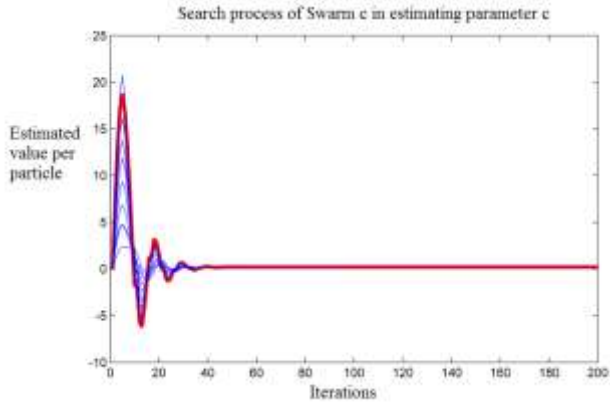


Figure 13: Search process of Swarm c

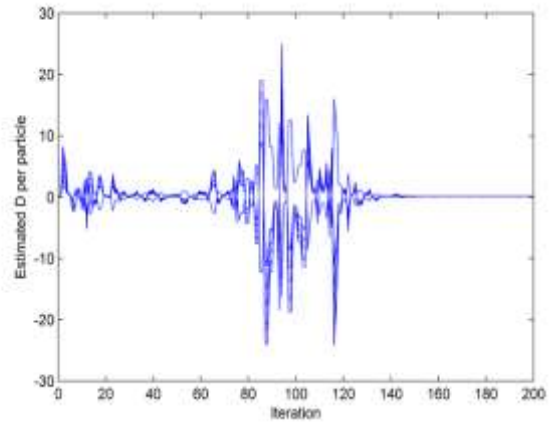


Figure 16: Search process of particles in PSO during estimating parameter d

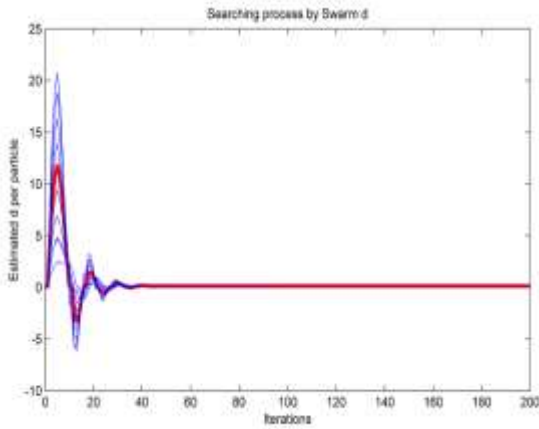


Figure 14: Search process of Swarm d

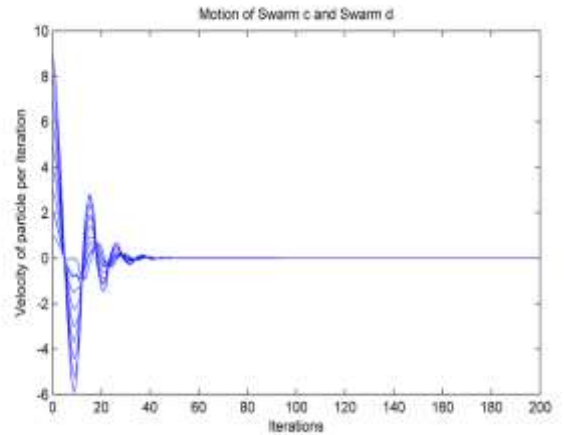


Figure 17: Velocity of Swarm c and swarm d in CMS-PSO

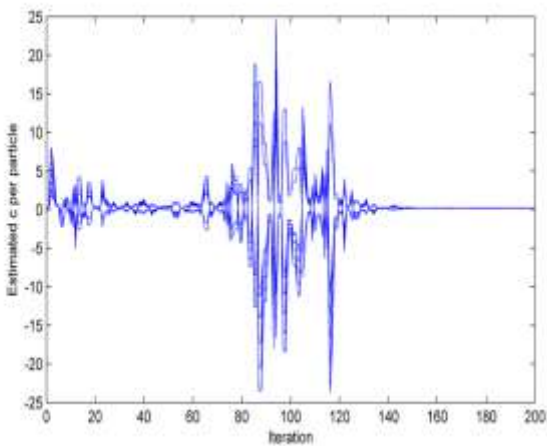


Figure 15: Search process of particles in PSO during estimating parameter c

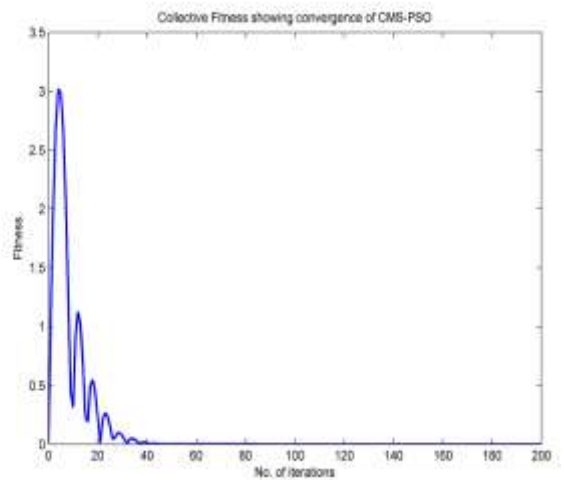


Figure 18: : Global Fitness fitness in estimating c and d parameter

It is observed from Fig.13 and Fig.14 that all the particles of the swarm delegated to estimate the parameters follow more or less the same pattern of searching process, converging successfully at, $\bar{c} = 0.200000$ and $\bar{d} = 0.120000$. The red line shows the path of the particle which converged first at the global position. All the particles communicate with each other, tuning their velocity to attain the optima. On closer examination, it is observed that when the global optimum is reached, then the particles stop moving in the search space as shown in Fig 17 with velocity gradually tending to zero. Another striking observation is the spatial symmetry in the searching pattern of the swarms. The similar pattern obtained can be attributed to the chaotic sequences which has substituted the random number generator. The chaotic sequences although being disordered follow an orderly fashion bringing faster convergence with the optimized result. This can be validated from the highly fluctuating and disordered trajectory exhibited by the PSO in Fig 15 and Fig. 16. Moreover, all the particles do not converge faster in comparison to CMS-PSO. This highlights the fact that concurrently running and cooperating cluster of particles can infact govern the behavior of the dependent swarm. This is akin to a master-slave relationship wherein the behavior and system parameters of the response system are dependent on the behavior of the driving system.

Fitness of the particles is reflected from Fig 18 which shows that the composite fitness of the swarms when estimating c and d approaches ϵ gradually, which is zero in our result.

4.2 Application in Global Optimization and conclusion

Let us consider a system

$$\dot{x}_k = f(x_k, \theta) \text{ where } k=1,2,3,\dots,n$$

where x_k are the state variables and $\theta = (\theta_1, \theta_2, \dots, \theta_d)$ are the systems parameters. The estimated system is

$$\dot{y}_k = F(y_k, \bar{\theta}) \text{ where } k=1,2,3,\dots,n$$

To estimate the parameters simultaneously we need to construct the fitness functions corresponding to every parameters. We now explain this process step wise.

Step -1: For any arbitrary parameter say $\bar{\theta}_p$ being estimated,

CMS-PSO dynamically forms a swarm $S_{\bar{\theta}_p}$ where $p=1,2,\dots$

.d. The corresponding fitness function $J_{\bar{\theta}_p}$ depends upon the equations containing the parameter θ_p .

Case-A: If there is exactly one equation say $[\dot{x}_m = f_m(x, \theta)]$

where $m \in n$, that contains the parameter θ_p only, together with other $(\theta - \theta_p)$ parameters, then the corresponding

$J_{\bar{\theta}_p}$ is of the form

$$J_{\bar{\theta}_p} = \text{absolute} [f_m(x, \theta) - \dot{y}_m]$$

Case-B: If there are k equations containing the parameter θ_p ,

then the corresponding $J_{\bar{\theta}_p}$ is of the form

$$J_{\bar{\theta}_p} = \sqrt{\left(\sum_{g=1}^k \| [f_g(x, \theta) - F_g(\dot{y}, \bar{\theta}_p)] \|^2\right)}$$

Step-2 : After constructing all the fitness function $J_{\bar{\theta}_p}$,

$p=1,2,\dots,d$, we will run CMS-PSO to estimate all the parameters together using eq (7) and (8). The searching process can be archived if all the fitness functions are minimized with a desired accuracy.

Note-1.It may occur that two fitness function $J_{\bar{\theta}_p}$ and

$J_{\bar{\theta}_q}$ are are same ($p,q=1,2,\dots,d$)

Note-2. At first step all the parameters will take the values from the initial choice. From the next iterations, the objective law will be updated through our modified PSO algorithm and the swarms evolve to reach the steady state.

We now illustrate the above constructs with a small example.

To estimate the parameters a, b, c, d. There are 3 instances of parameter a appearing in eq 9, 10 and 11 and eq (12) whereas parameter b appears in only one equation (eq 11). Parameter c and d also appear in one equation 12. So four swarms are formed

dynamically for the four estimated parameters viz $S_{\bar{a}}$,

$S_{\bar{b}}$, $S_{\bar{c}}$ and $S_{\bar{d}}$. The corresponding estimated system would take the form

$$\dot{y}_1 = F(y_1, y_2, y_3, y_4) = S_{\bar{a}} (y_1 + y_2) + y_4 \quad (9.1)$$

$$\dot{y}_2 = F(y_1, y_2, y_3, y_4) = -y_2 - S_{\bar{a}} y_1 y_3 \quad (10.1)$$

$$\dot{y}_3 = F(y_1, y_2, y_3, y_4) = S_{\bar{b}} + S_{\bar{a}} y_1 y_2 \quad (11.1)$$

$$\dot{y}_4 = F(y_1, y_2, y_3, y_4) = S_{\bar{c}} y_1 y_3 - S_{\bar{d}} y_4 \quad (12.1)$$

For each particle $i=1,2,\dots,s$ at some instant t, the CMS-PSO evolves after an initial state y_0 with the initial random values of velocity and random values of \bar{a} , \bar{b} , \bar{c} , \bar{d} in the desired range. The corresponding fitness function are constructed as

$$J_{\bar{a}}[i] = \sqrt{(f_1[i] - F_1[i])^2 + (f_2[i] - F_2[i])^2 + (f_3[i] - F_3[i])^2}$$

$$J_{\bar{b}}[i] = \text{absolute}(f_3[i] - F_3[i])$$

$$J_{\bar{c}}[i] = J_{\bar{d}}[i] = \text{absolute}(f_4[i] - F_4[i])$$

From above, the values of i th particle at t instant in the swarm $S_{\bar{b}}$ are purged into $S_{\bar{a}}$. Similarly estimated values by particles in $S_{\bar{c}}$ are used by $S_{\bar{d}}$ to estimate d . Global optimization occurs when summation of all individual objective function J is less than ϵ i.e

$$\frac{1}{n} \sum J_{\bar{\theta}_p} \leq \epsilon$$

Thus the actual parameters are fully identified and the objective function converges to zero.

The proposed CMS-PSO is a robust and decentralized option for dynamical system, easily implemented in a computationally inexpensive manner, since its memory and CPU speed requirements are low. It requires only the values of the gradient and no gradient information of the objective function is needed to obtain the pareto optimal solution. The multiple swarms handle the optimization task which communicate and co-exist with the other swarms governed by rules of mutual co-existence similar to ecological systems. CMS-PSO addresses the problem of premature convergence and enhances the scope of navigation in an orderly fashion due to the distinct inherent properties of chaotic sequence. CMS-PSO dynamically spawns swarms in accordance to the global optimization problem domain. The different cluster of swarms for every state vector cooperate and co-exist to balance the global optimization task by dividing the objective function into sub-parts and delegating each swarm to yield the precise optimal result for each parameter. In effect we are dealing with a dynamical system for multi objective parameter optimization.

The results are useful from the perspective of management for prediction and estimation of different parameters from a chaotic/hyperchaotic economic models from its time series. In the area of finance, stocks and social economics, due to the interaction between nonlinear factors, with the evolution process from low dimensions to high dimensions, there exists extremely complicated phenomenon and external characteristics in such a kind of system. Our system is a mathematical model of Finance, so it has become imperative to make a systematic and deep study of the internal structural characteristics of the nonlinear system. A detailed analysis of the nature of the system has been attempted by numerical simulations of the model and its data series analysis. CMS-PSO can be applied in control theory since a system which is chaotic in nature can revert back to a steady state.

REFERENCES

[1] de Castro L.N., Timmis, J. (2002). Artificial Immune Systems: A New Computational Intelligence Approach. Springer. ISBN 1852335947, 9781852335946: 57–58.

- [2] Kephart, J. O. (1994) A biologically inspired immune system for computers. Proceedings of Artificial Life IV: The Fourth International Workshop on the Synthesis and Simulation of Living Systems. MIT Press: 130–139.
- [3] Haykin S. (1999). Neural Networks: A Comprehensive Foundation, Prentice Hall, ISBN 0-13-273350-1.
- [4] Zadeh L.A. (1965). Fuzzy sets, Information and Control **8** (3): 338–353.
- [5] Abraham A. (2005) Evolutionary Computation, In: Handbook for Measurement, Systems Design, Peter Sydenham and Richard Thorn (Eds.). John Wiley and Sons Ltd. London. ISBN 0-470-02143-8: 920-931.
- [6] Rechenberg, I. (1973) Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution, Stuttgart: Fromman-Holzboog.
- [7] Schwefel, H.P. (1977). Numerische Optimierung von Computermodellen mittels der Evolutionsstrategie, Basel: Birkhaeuser.
- [8] Fogel, L.J., Owens, A.J. and Walsh, M.J. (1966) Artificial Intelligence Through Simulated Evolution. John Wiley and Sons Inc. USA
- [9] Holland, J.H. (1975). Adaptation in natural and artificial systems. The University of Michigan Press Ann Arbor.
- [10] Koza, J.R. (1992). Genetic Programming : On the Programming of computers by Means of natural selection : The MIT Press Cambridge MA
- [11] Kennedy, J. and Eberhart, R.C. (1995) Particle swarm optimization. Proc IEEE Int Conf on Neural Networks, Piscataway:1942–1948.
- [12] Kennedy, J. and Eberhart, R.C. and Shi, Y. (2001) Swarm intelligence. San Francisco, CA: Morgan Kaufman.
- [13] Zinkova, M(Author). (2004) School of Goldband Fusilier, *Pterocasio chrysozona*. Photograph taken in Papua New Guinea. Wikimedia Commons.
- [14] Yann(Author) (2010). Bee swarms. Jaura MP India.. Wikimedia Commons.
- [15] Jha, S and Roy, D(Author). (10 January 2010). The Heat and Dust Project (travelogue - work in progress). Photograph taken in Jaisalmer, Rajasthan. Pub: HarperCollins India. (Rights of reproduction reserved) .
- [16] Parsopoulos, K.E. and Vrahatis, M.N. (2002). Recent approaches to global optimization problems through Particle Swarm Optimization. Natural Computing **1** : 235–306.
- [17] Grassberger, P. and Procaccia, I. (1983). Measuring the strangeness of strange attractors. Physica D **9** :189-208.
- [18] L'Ecuyer P. (1996) . Combined Multiple Recursive Random Number Generators, Operations Research **44**(**5**): 816–822.
- [19] Thomaidis, N., Angelidis, T., Vassiliadis, V. and Dounias, G. (2009), Active Portfolio Management With Cardinality Constraints: An Application Of Particle Swarm Optimization,

New Mathematics and Natural Computation (NMNC). World Scientific Publishing Co. Pte. Ltd. **5(03)** : 535-555.

- [20] Marinakis, Y., Marinaki, M., Doumpos, M. and Zopounidis, C. (2009). Ant colony and particle swarm optimization for financial classification problems, *Expert Systems with Applications* **36(7)** : 10604-10611.
- [21] Jha, G.K, Kumar, S., Prasain, H., Thulasiraman, P. and Thulasiraman, R.K. (2009). Option pricing using Particle Swarm Optimization : *Proceedings of the 2nd Canadian Conference on Computer Science and Software Engineering* : 267-272 .
- [22] Bibby, B.M. and Sørensen, M. (1995). Martingale estimation functions for discretely observed diffusion processes, *Bernoulli* **1(1/2)** : 017-039.
- [23] He, Q., Wang, L. and Liu, B.(2007). Parameter estimation for chaotic systems by particle swarm optimization. *Chaos, Solitons and Fractals*, **34** : 654–661.
- [24] Tang, Y. and Guan, X. (2009). Parameter estimation for time-delay chaotic system by particle swarm optimization. *Chaos, Solitons and Fractals*, **40** : 1391–1398.
- [25] Liang, J.J. and Suganthan, P.N. (2005). Dynamic multi-swarm particle swarm optimizer, *Swarm Intelligence Symposium, Proceedings IEEE* : 124 – 129.
- [26] Sun, J., Zhao, J., Wu, X., Fang, W., Cai, Y. and Xu, W. (2010). Parameter estimation for chaotic systems with a Drift Particle Swarm Optimization method. *Physics Letters A*.
- [27] Fan, S.K.S. and Chang, Ju-Ming. (2010). Dynamic multi-swarm particle swarm optimizer using parallel PC cluster systems for global optimization of large-scale multimodal functions, *Engineering Optimization* **42(5)**: 431 – 451.
- [28] Shannon, C.E. (1949) Communication theory of secrecy systems. *Bell Syst. Tech. J* **28**: 656-715.
- [29] Caponetto, R., Fortuna, L., Fazzino, S. and Xibilia, M.G. (2003). Chaotic sequences to improve the performance of evolutionary algorithms, *IEEE Transactions on Evolutionary Computation*; **Vol 7 (3)** : 289-304.
- [30] Coelho, L.D.S. and Mariani, V.C. (2008). Use of chaotic sequences in a biologically inspired algorithm for engineering design optimization. *Expert Systems with Applications* **34**:1905–1913.
- [31] Shi, Y. and Eberthart, R.C.(1998). A modified particle swarm optimizer. *Proc IEEE Int Conf on Evolutionary Computation*, Anchorage :69–73.
- [32] Li YX, Tang WKS, Chen G. Generating Hyperchaos via State Feedback Control. *Int. J. Bifurcat. Chao.*2005; **15(10)** : 3367-3375.
- [33] Ma, J.H. and Chen, Y.S.(2001). Study for the bifurcation topological structure and the global complicated character of a kind of nonlinear finance system (I). *Appl. Math. Mech* **22**: 1240-1251 . (*Englished.*)
- [34] Ma, J.H. and Chen, Y.S.(2001). Study for the bifurcation topological structure and the global complicated character of a kind of nonlinear finance system (II). *Appl. Math. Mech.* **22**: 1375-1382 (*Englished.*).
- [35] Ma, J.H. and Chen, Y.S. (2004). Impulsive control of chaotic Attractors in Nonlinear Chaotic systems. *Appl.Math. Mech.*; 25:889-894.